Machine Learning for Space Projects:
Example Engineering and Science Case Studies

Hamed Valizadegan

Data Science Group @ NASA Ames Research Center

Machine Learning

- Promise: Give me Data, I return knowledge/information/expertise
- It is the science of learning from data
- Consists of optimization + probability + many task/data specific tricks such as pre-processing/post-processing/visualization/etc.
- Often, a task specific, data dependent objective function is written and optimized
  - In many problems, the learning is nothing but optimizing an objective function to obtain the value of unknown variables
  - Objective function to optimize is data dependent
  - Values of these variables are the knowledge/information/expertise
  - Will provide an example to discuss this further

Machine Learning (Cont'd)

- It is also an art (like programming)
  - Usually starts with staring at the data (visualizing data helps a lot)
  - There is a design element
  - Each application and data demand a specific treatment decided by the machine learning expert
  - Many different tools and techniques are available.
  - For the same data/task, two different machine learning experts may take two different approaches
  - The correct usage of each technique (from implementation to validation) needs knowledge and experience

- Process: have an idea, try it, learn from it, come up with a new idea, …
  - There are always ways to improve: more data, better data, better learning algorithms, better preprocessing, etc
  - Will present two case studies and their (ongoing) machine learning processes

Machine Learning for Space Problems

- Example applications
  - morphological classification of galaxies, classification of asteroids, Star/Galaxy classification
  - photometric redshift of galaxies (Regression)
  - Anomaly detection in Space related instruments (Space Shuttle Main Propulsion System)
  - Anomaly detection in astronomical observations (unusual single star or single galaxy, unusual cluster of galaxy)

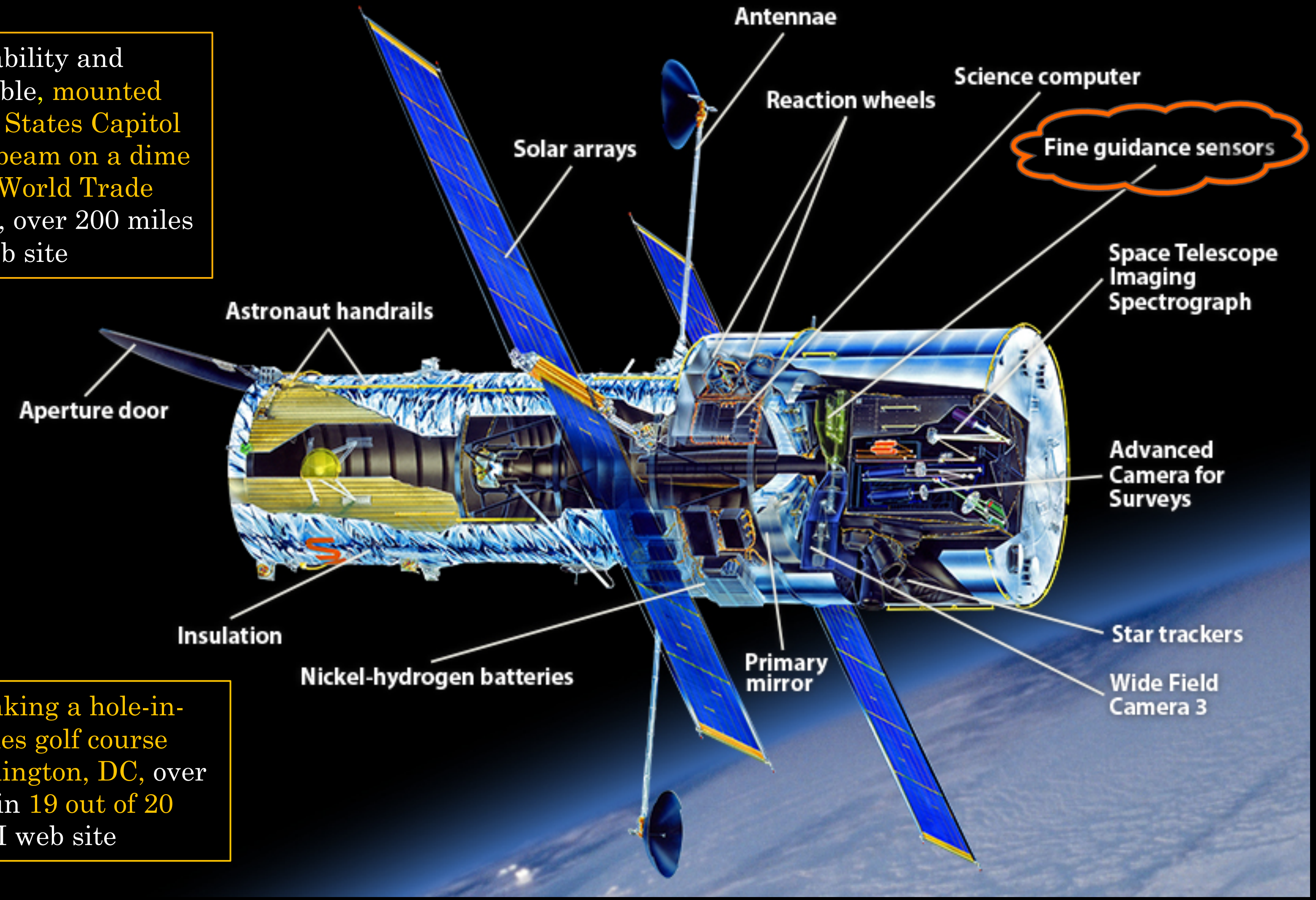
- Engineering vs Science Problems
  - Engineering domain: Helps to run or maintain the space instrument better
    - I will discuss the lifetime prediction of Hubble Space Telescope as an example
  - Science domain: increase our knowledge of the space
    - I will discuss our efforts in the automatic discovery of exo-planet using Kepler mission data.

	Sample Size	Labels	Domain Knowledge	Relevant literature
Engineering	Very small (extreme)	limited	Complex, limited	limited
Science	medium…big	Limited, inaccurate	Limited (unknown objects)	limited

Hubble Space Telescope (HST)

- Low earth orbit space telescope, lunched in Apr 1990, and still in operation
- provided us with spectacular images of the universe and increased our understanding of the universe significantly
- Repair/upgrade/replacements have been done over five servicing missions in Dec 1993, Feb 1997, Dec 1999, Mar 2002, May 2009.
- An independent team at NASA was built to study the remaining useful lifetime of HST (NESC)
- Most components of HST was studied using life-usage model or hand-coded physics-based prognostics
- The complexity of one component, Fine Guidance Sensor (FGS), made life-usage model or hand-coded physics based model infeasible

“A laser with the stability and precision of the Hubble, mounted on top of the United States Capitol could hold a steady beam on a dime suspended over the World Trade Center in New York, over 200 miles distant.” –STSCI web site



“Comparable to sinking a hole-in-one on a Los Angeles golf course from a tee in Washington, DC, over 2,000 miles away, in 19 out of 20 attempts.” – STSCI web site

Fine Guidance Sensor (FGSs) at HST

Three FGSs exists in HST, responsible (1) to keep the telescope accurately pointed at a target, and (2) to act as a science instrument to measure the brightness and relative positions of stars (Astrometry).

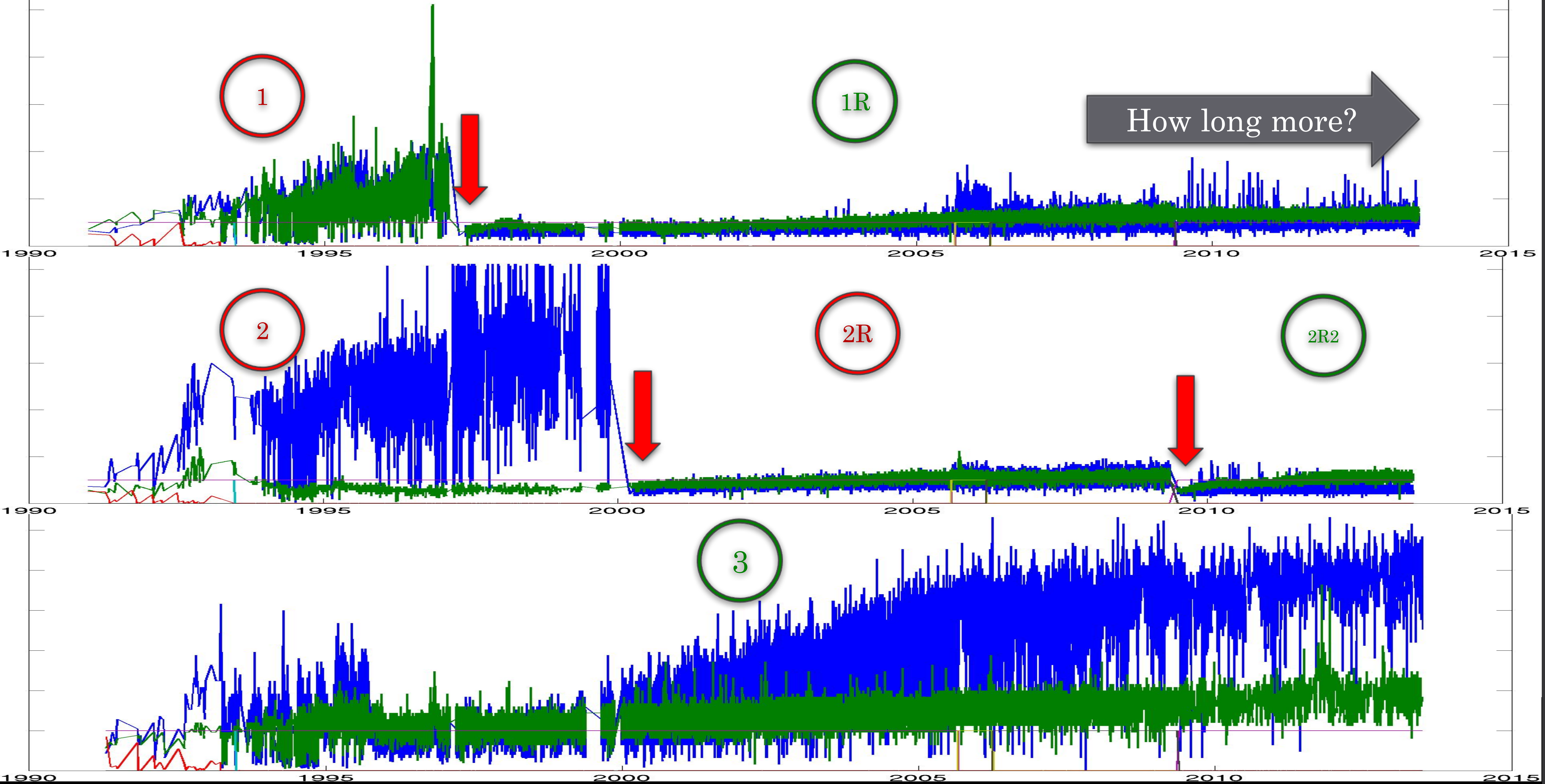
“Hubble is the most precisely pointed machine ever devised for astronomy. The telescope must be able to maintain lock on a target for 24 hours without deviating more than 7/1,000ths (0.007) of an arc second (2 millionths of a degree) which is about the width of a human hair seen at a distance of a mile.”

Predicting lifetime of FSGs

Contributors: Bryan Matthews, Koushik Datta, & other NESC team

- The units were build specifically for HST and for the first time
- No mass production
  - life usage model is not applicable
- The domain knowledge was very limited and the units were complex
  - Hand-coded physics-based prognostics was not accessible
- Data driven approach is very challenging
  - Extreme sample size problem
  - few failed and working units
- Objective: given the collection of sensory data of the current and failed FGS units, build a model to predict the remaining lifetime of current units
  - We developed two methodologies to study remaining lifetime/degradation of these units

One of the three Fine Guidance Sensors photographed during Second Servicing Mission in 1997



FGS data: a close look

Each row shows one of the FSGs at HST. Red arrows shows the FGS was replaced

Predicting lifetime of FSGs

- Assume: remaining-lifetime of each unit is known
  - A regression problem (map the sensory data to remaining life-time values)
  - Y=F(W,X), W is unknown
- Reality: Unknown remaining lifetime
  - a regression problem with target values as variables to optimize
  - Y=F(W,X), Y and W are unknown
  - The trivial solution with all target values being zero: not interesting
- However, restrictions on the target values are available
  - Known: Remaining lifetime of failed units are small near the time they failed
  - Known: Remaining lifetime decrease over time (degradation increases)
  - Y=F(W,X), W and some Y are unknown. Some Y are known and there are constraints on others
  - Some sorts of semi-supervised learning (regression)

Term for failed units (m=1,2,3): known target values

$$\min_{\mathbf{w},\{\mathbf{y}_i,i=m+1,\ldots,N\}} \quad \|\mathbf{w}\|^2 + \beta_1 \sum_{i=1}^{m}\sum_{t=1}^{T_i} \|\mathbf{w}'\mathbf{x}_i^t - y_i^t\|^2$$

$$+ \beta_2 \sum_{i=m+1}^{N}\sum_{t=1}^{T_i} \|\mathbf{w}'\mathbf{x}_i^t - y_i^t\|^2 + \tau \sum_{i=m+1}^{N}\sum_{t=2}^{T_i} \epsilon_i^t$$

Term for working units (m=4,5,6): unknown target values

$$\begin{aligned} s.t. \quad & y_i^t \geq 1 \quad i = m+1,\ldots,N; \quad t = 1,\ldots,T_i \\ & y_i^t - y_i^{t-1} + 1 \leq \epsilon_i^t \quad i = m+1,\ldots,N; \quad t = 2,\ldots,T_i \\ & \epsilon_i^t \geq 0 \quad i = m+1,\ldots,N; \quad t = 2,\ldots,T_i \end{aligned}$$

The remaining lifetime at time t is less than that at time t-1

Linear model (can be extended to non-linear using kernel trick)

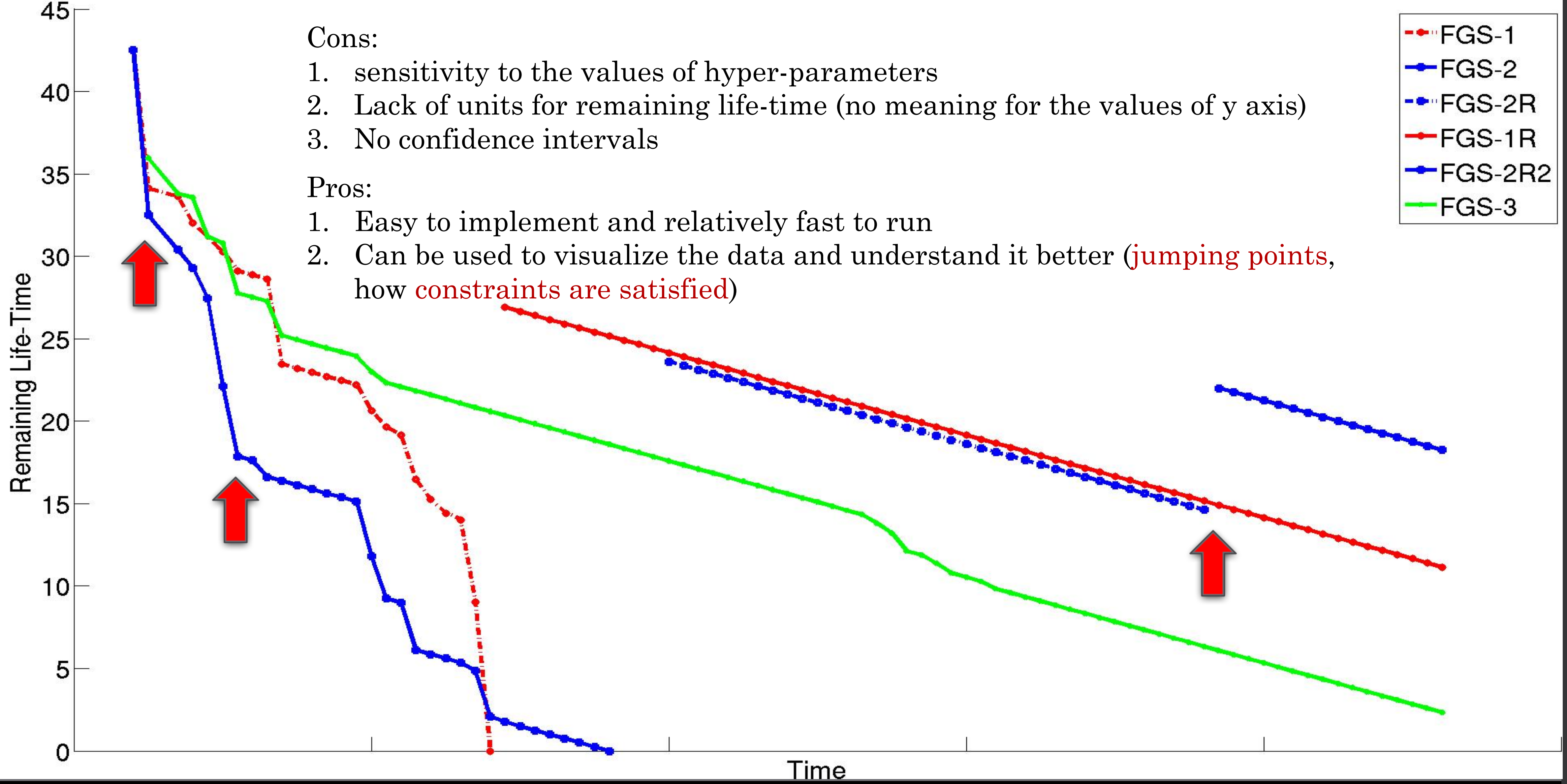
- Consider this as a regression problem in which values of output are unknown but constraints on output values are provided.
- In quadratic programming form and can be solved efficiently

Cons:

1. sensitivity to the values of hyper-parameters
2. Lack of units for remaining life-time (no meaning for the values of y axis)
3. No confidence intervals

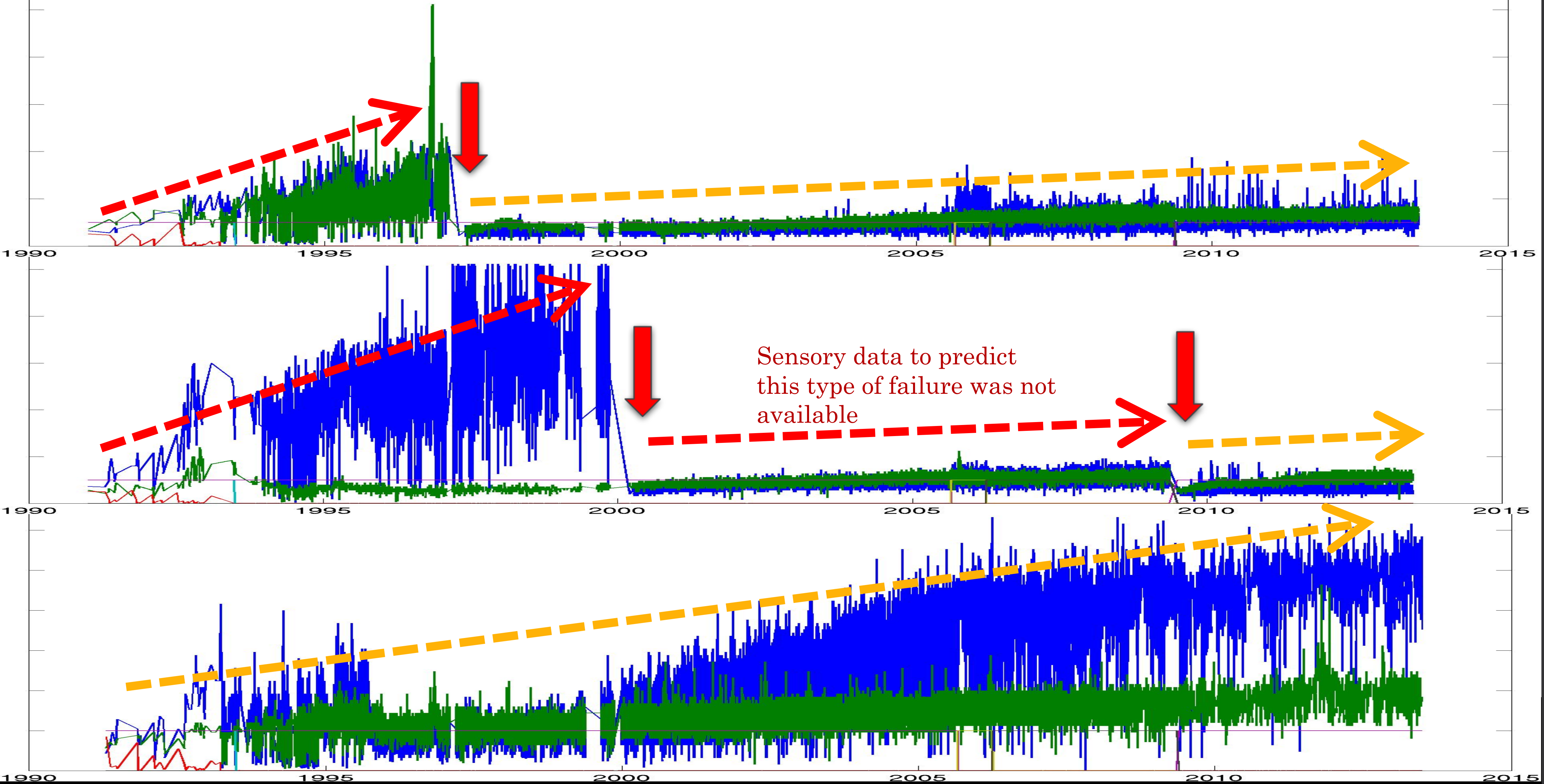
Pros:

1. Easy to implement and relatively fast to run
2. Can be used to visualize the data and understand it better (jumping points, how constraints are satisfied)



FGSs Degradation progress (remaining lifetime)

Color Code: One slot of FGS, for example blue shoes three FGS was used in slot 2

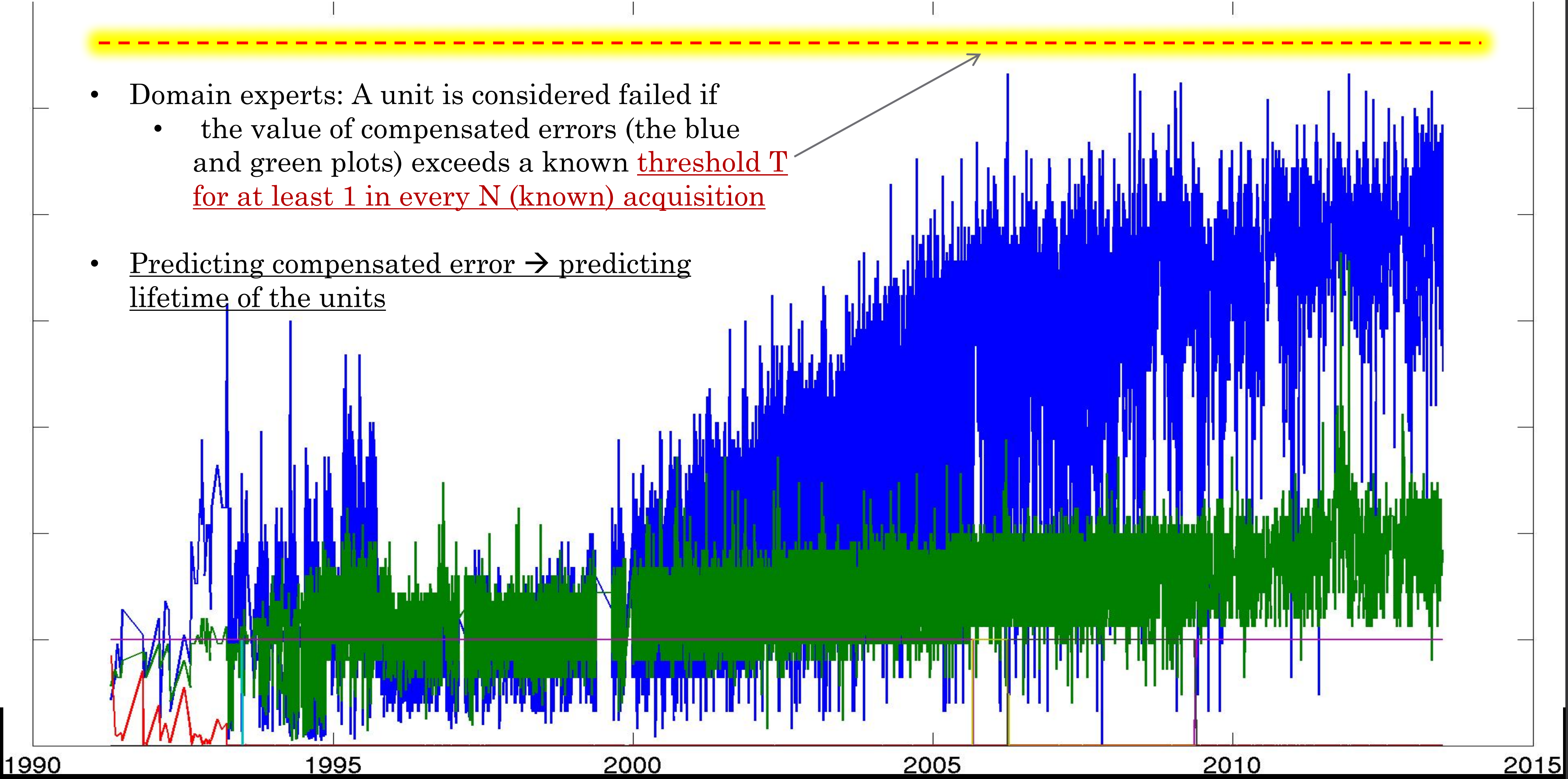


FGS data: a close look

Each row shows one of the FSGs at HST. Red arrows shows the FGS was replaced

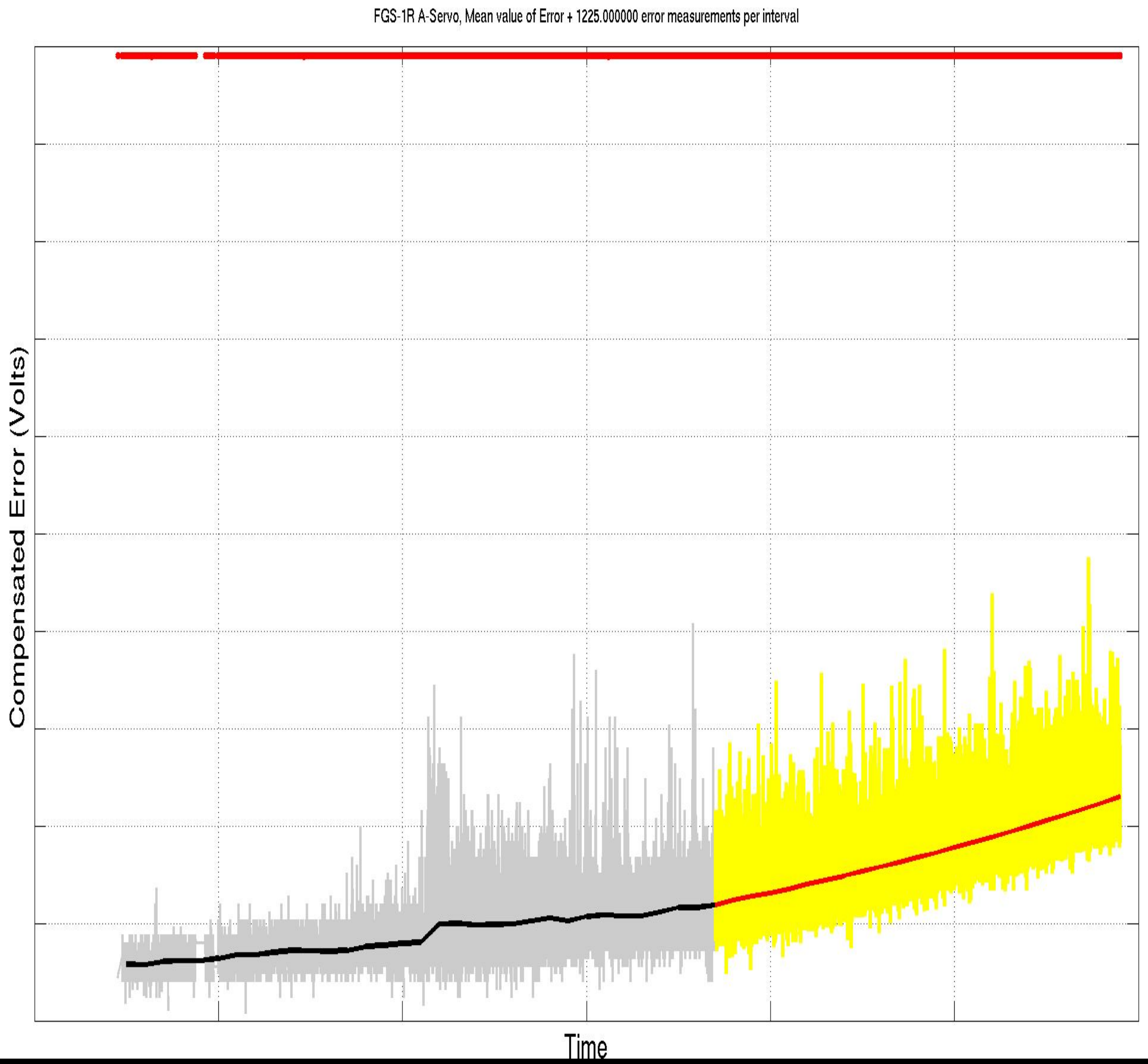
- Domain experts: A unit is considered failed if
  - the value of compensated errors (the blue and green plots) exceeds a known <u>threshold T for at least 1 in every N (known) acquisition</u>

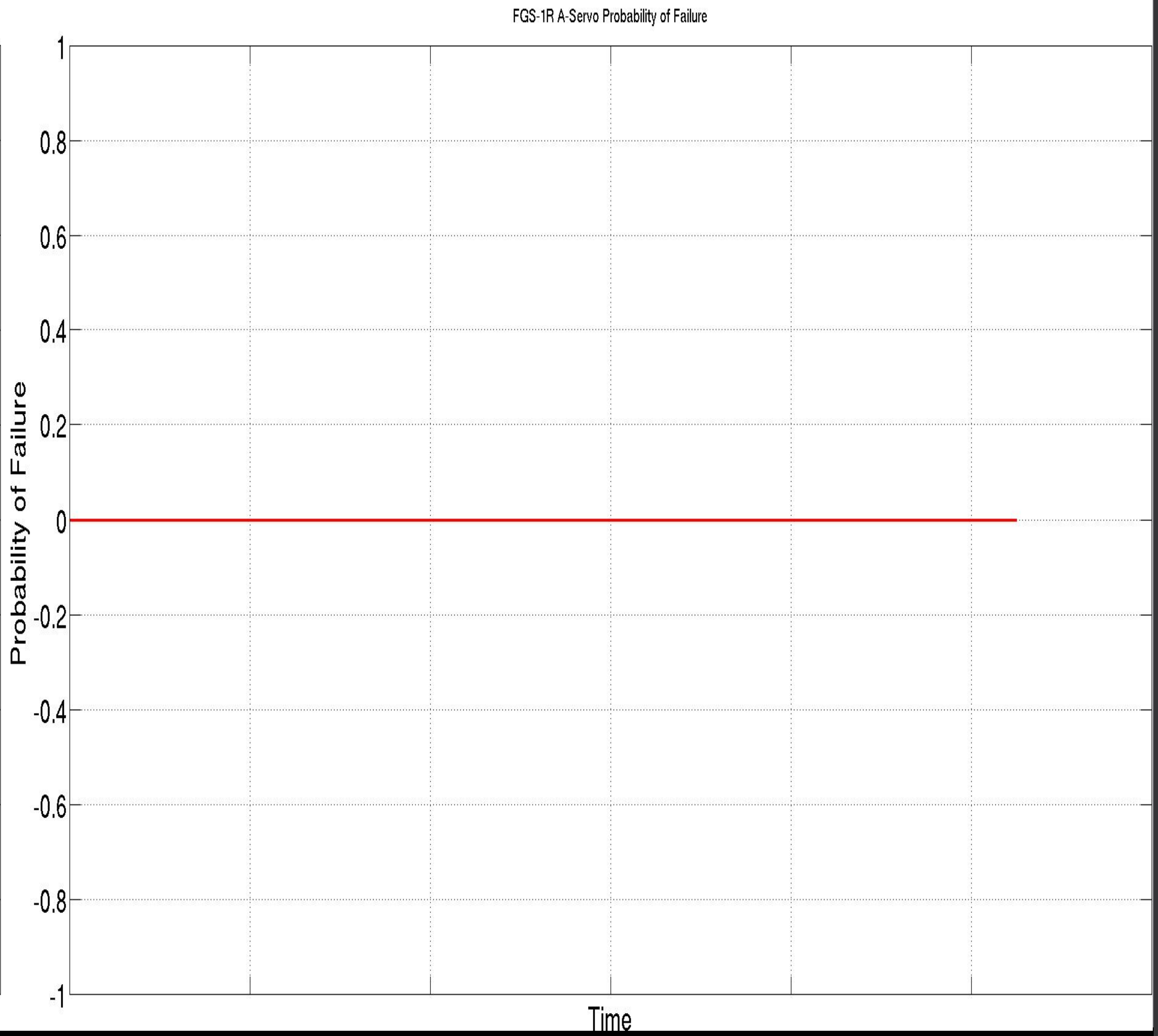
- <u>Predicting compensated error → predicting lifetime of the units</u>



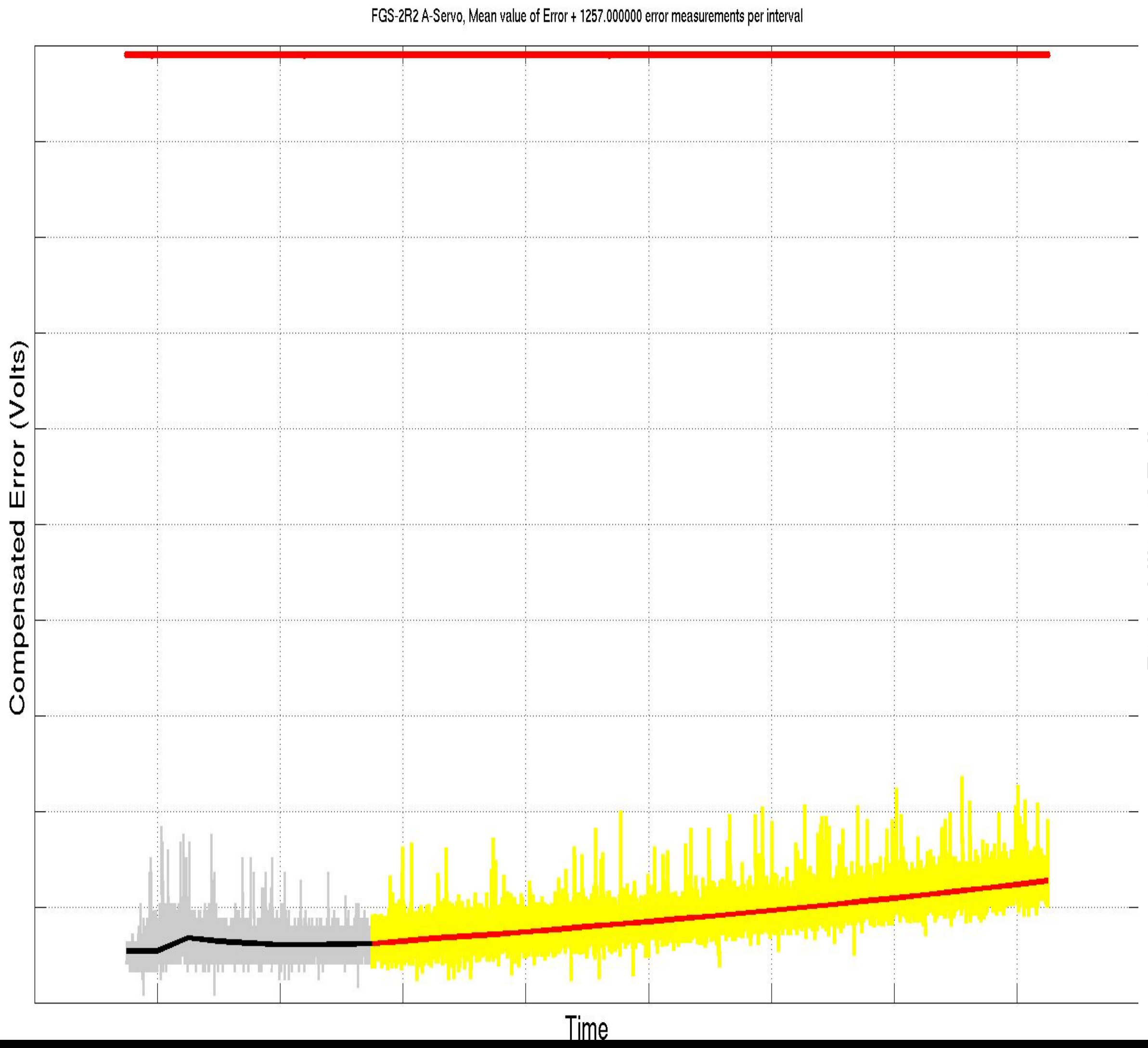
what domain experts say

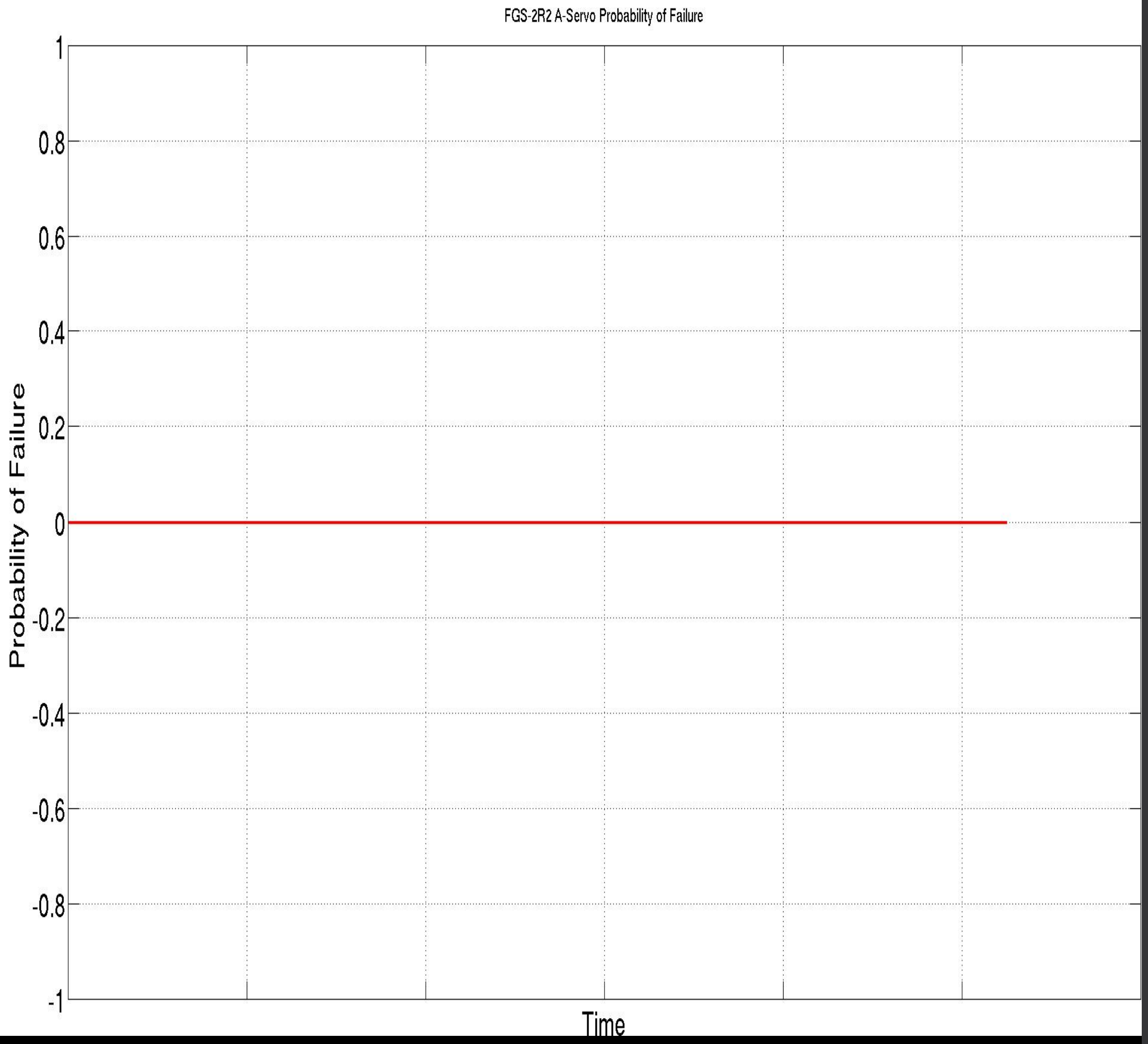
A simple information from domain experts makes the problem much easier.



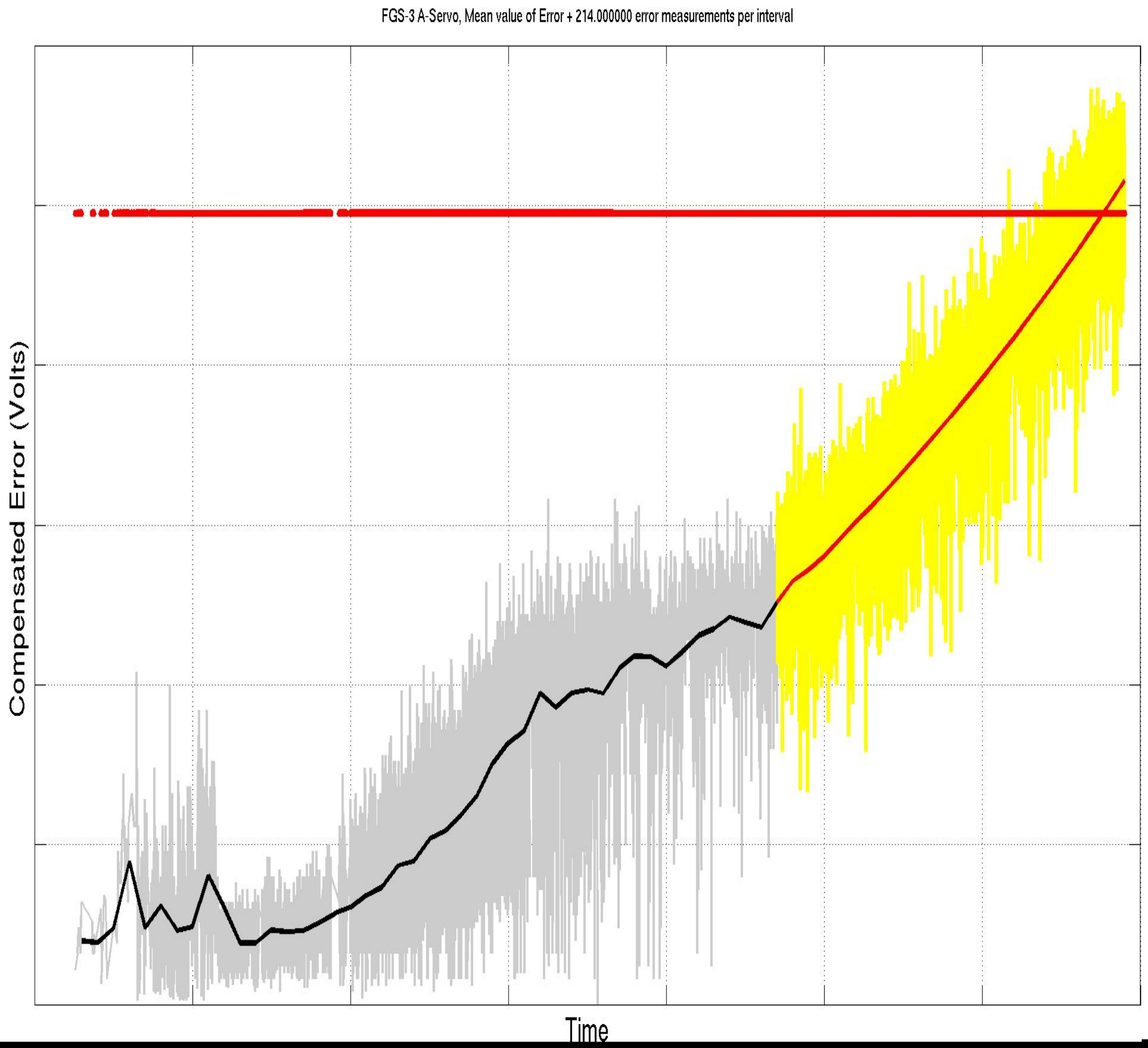


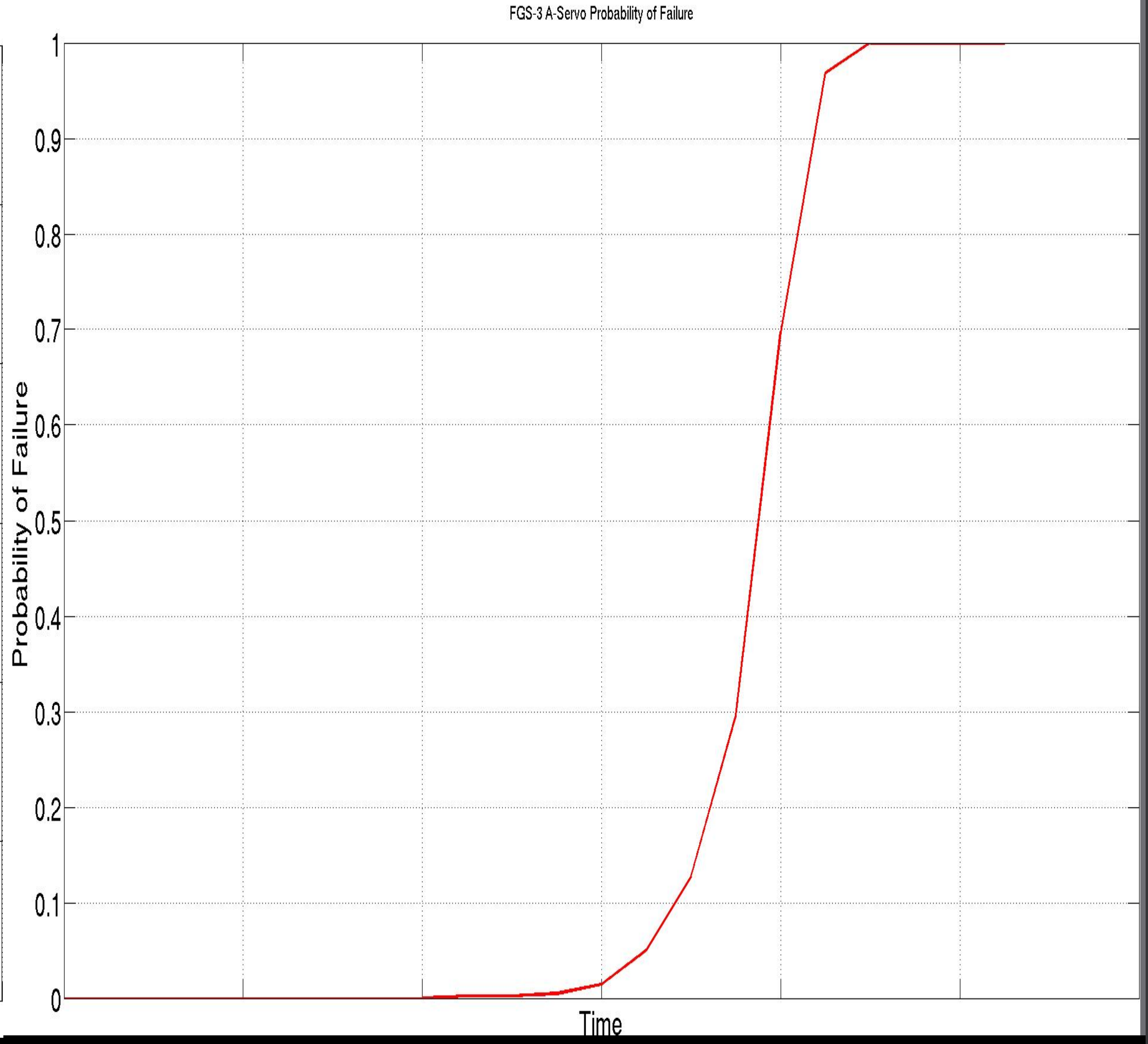
FGS 1R A-Servo: Error prediction and probability of failure





FGS 2R2 A-Servo: Error prediction and probability of failure



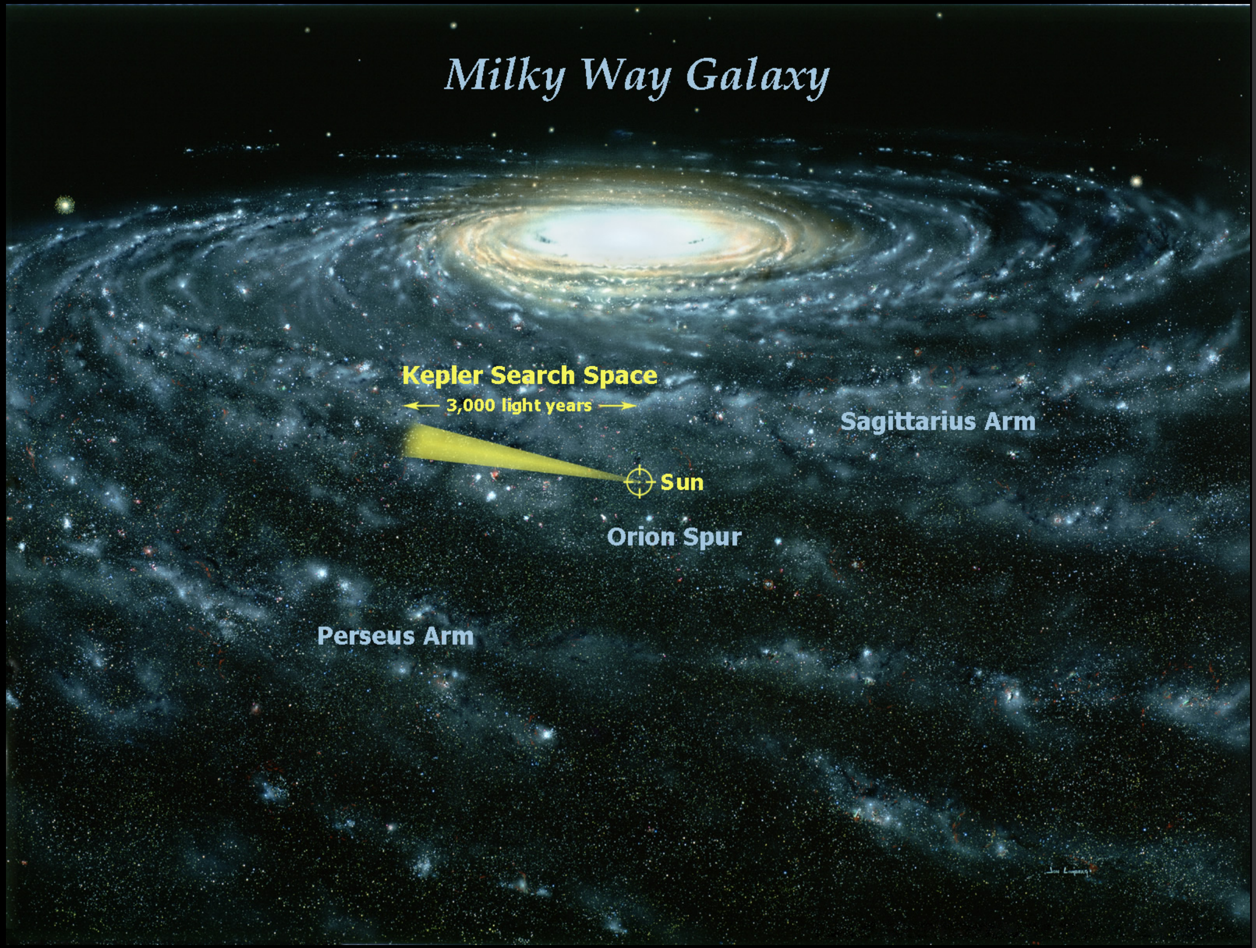


FGS 3 A-Servo: Error prediction and probability of failure

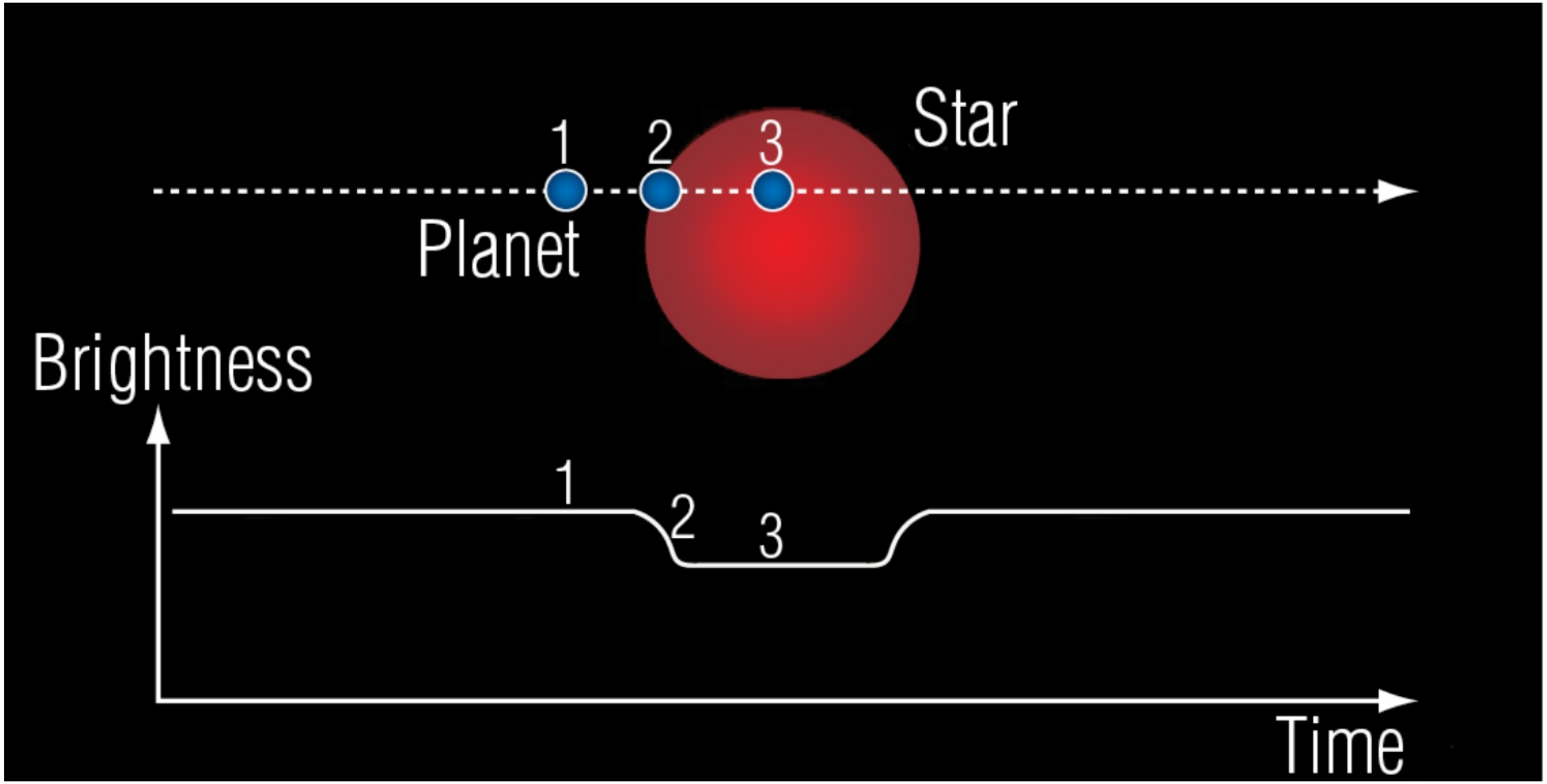
Domain knowledge can compensate for the small sample size

Kepler Space Telescope

- Designed to survey a small portion of milky way to discover earth-size planets or near habitable zone
- Lunched on March 2009
- In July 2012 and then May 2013, two of its reaction wheels used for pointing the spacecraft failed
    - Now it is being used for K2 extension mission
- More than 75% of all confirmed exoplanets (count:3422) are discovered by Kepler
- Using the knowledge gained from Kepler mission data
    - There are as many as 40 Billion rocky, earth-size planet with 11 billion of these planet orbiting sun-like stars

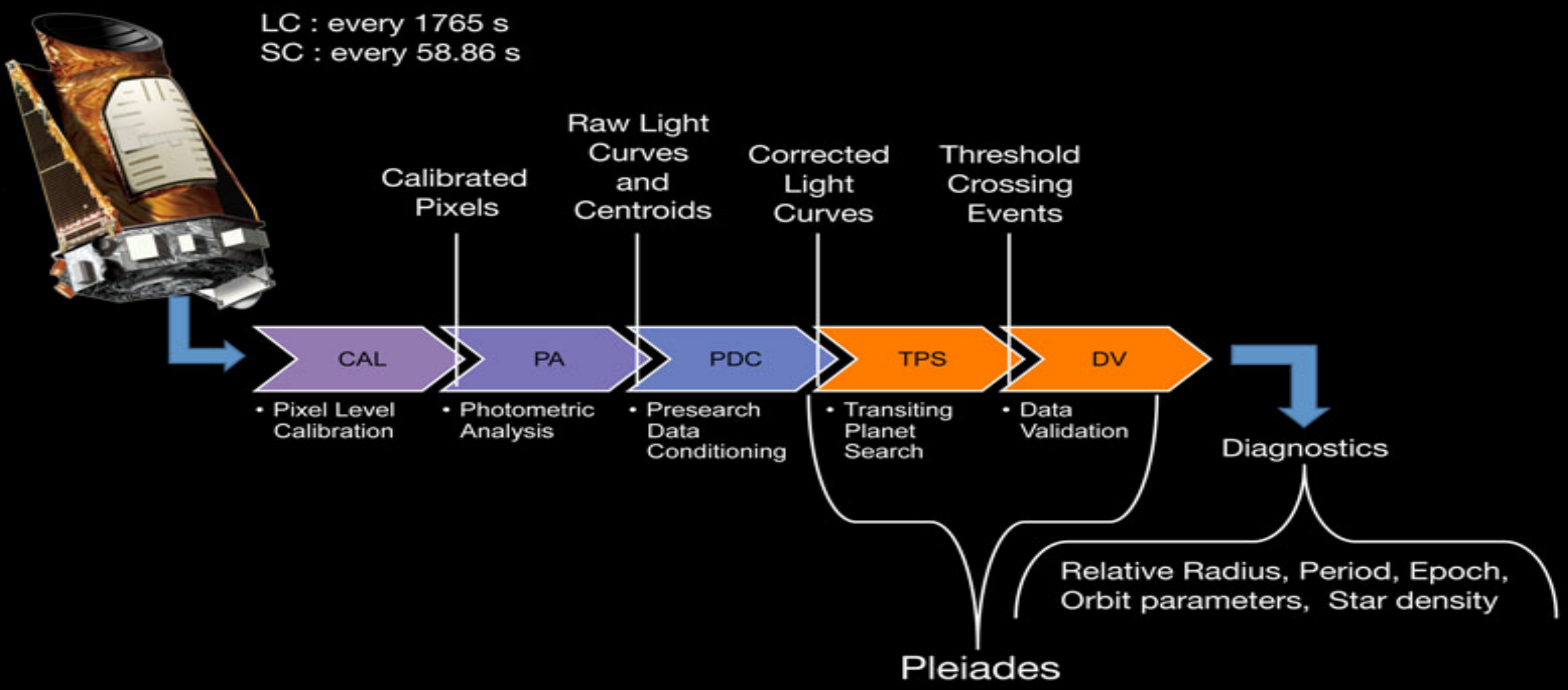


Transit Photometry



Three classes of interest

Planet Candidates (PC)
Astrophysical False Positive (AFP)
Non-Transiting Phenomena (NTP)



Kepler science processing pipeline

Pixels are downloaded once a month and transferred to the Science Operations Center (SOC), where they are calibrated, combined to form light curves, corrected for systematic errors introduced in the photometer, and then searched for the signatures of transiting planets. When a planet passes (or transits) in front of its host star, it blocks a small fraction of the light from that star that appears as tiny, repeating pulse or beat. By measuring the frequency of these beats and the amount of light blocked, we can detect the planets and calculate their size and orbital distance.

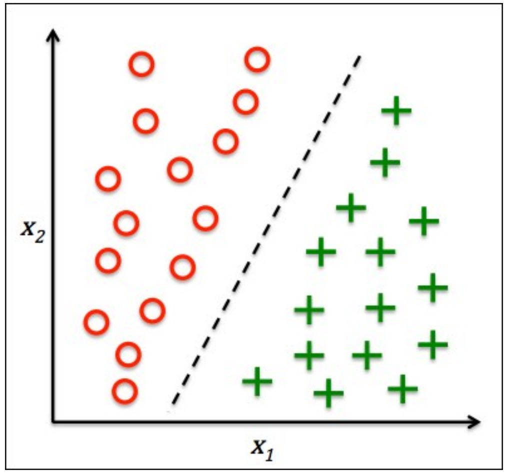
Classification of TCEs

Contributors: Jon Jenkins, Joe Catanzarite, Sean McCauliff

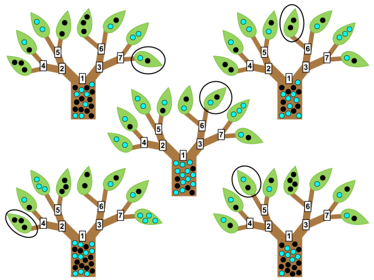
- Threshold crossing events (TCEs) are subjected to a vetting process performed by Kepler TCE Review Team (TCERT)
- Initial stage of vetting called triage: partitions the objects into
  - (1) problematic light curves that have instrumental noise and called Non-Transiting Phenomena (NTP), and
  - (2) Kepler Objects of Interests (KOI).
- KOIs are further scrutinized in later stages of vetting to be categorized to (1) Astrophysical False Positive (AFPs, e.g. Eclipsing Binary), and (2) Planetary Candidates (PC)
  - Basically: KOI= PC+AFP in this categorization
- Vetting process is challenging
  - Required a reviewing team of astrophysicist and astronomers
  - > 100 diagnostics metrics and associated graphics for each candidate exoplanet like signals
- Final objective: automating the process of classifying the TCEs into three classes: PC, AFP, and NTP (work in progress)

Machine Classification

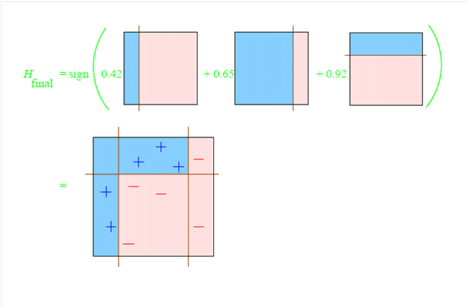
- Supervised learning
- Need a set of training cases (entities and their labels)
- Cases are usually represented by a feature vector extracted from a complex entity (e.g. graph or time series)
- A held-out set (test set) is used to measure how the learned model performs (generalization capability)



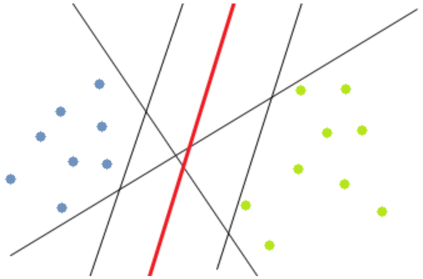
Random Forest



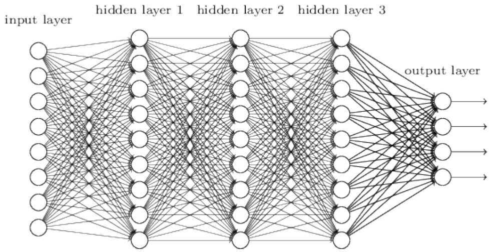
Gradient Boosting



SVM

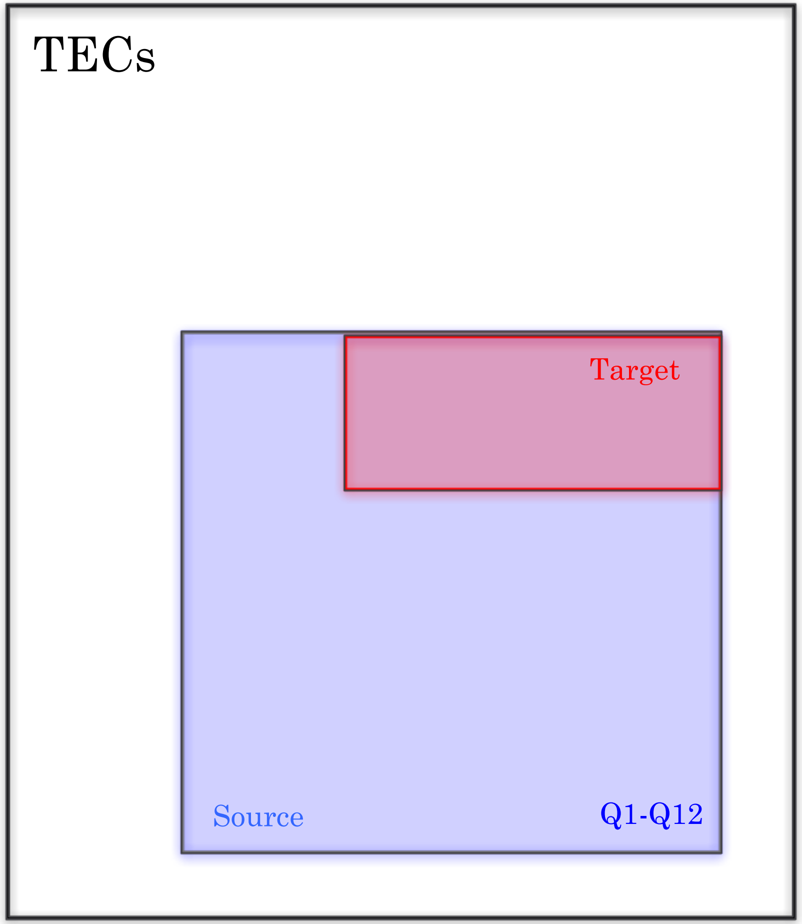


Neural Network (Deep)



Experiments on Q1-Q12

- 237 features (extracted from Q1-Q12, reduced to 216)
- 14576 objects (2879 PC, 393 AFP, and 11304 NTP), relatively accurately labeled
  - We call this source set
- 1487 have been given dispositions by the TCERT, (389 PC, 1098 AFP, and 0 NTP)
  - We call this target set
- Early machine classification results[1] (random forest):
  - When experimented on the source set, accuracy: ~99%
  - Poor result: model built from Source Set applied to Target set. Accuracy: 52%
  - Confirmed with different other models.



Source (Random Forest)			
TCERT vs Prediction	PC	AFP	NTP
PC	2843	8	28
AFP	98	271	24
NTP	25	12	11267

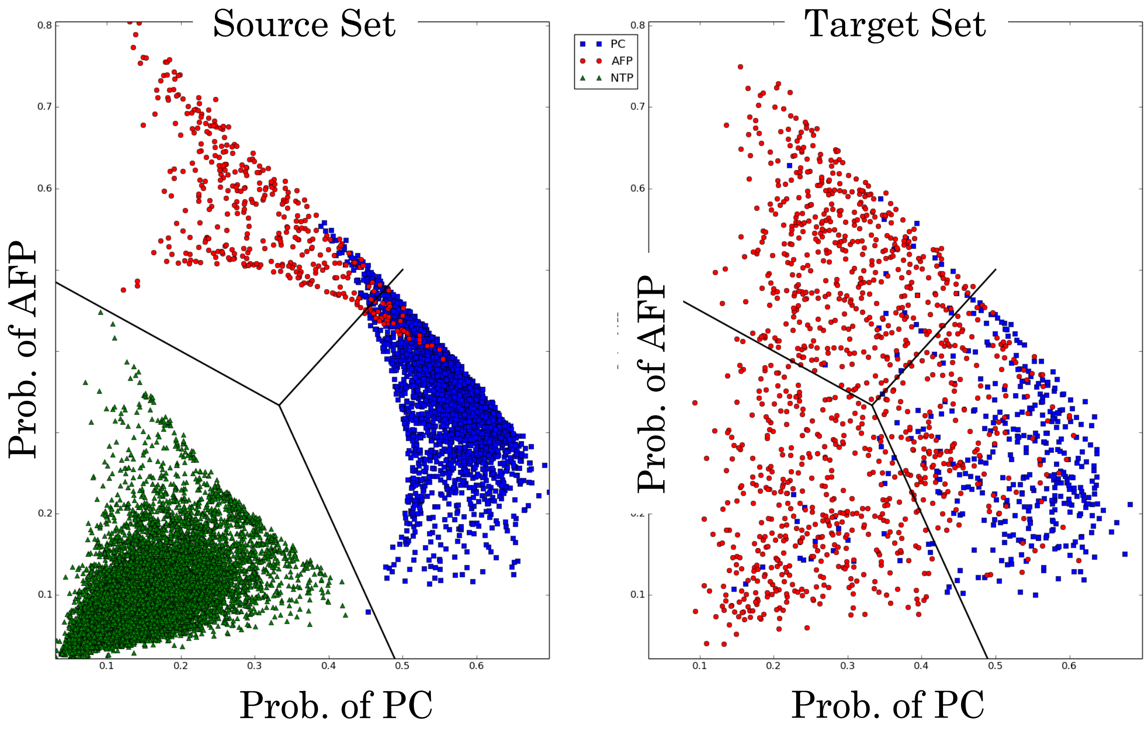
Target (Random Forest)			
TCERT vs Prediction	PC	AFP	NTP
PC	336	10	43
AFP	204	388	506
NTP			

1 S. McCauliff et al., Automatic Classification of Kepler Threshold Crossing Events, ApJ, 2014

Poor performance on target set?

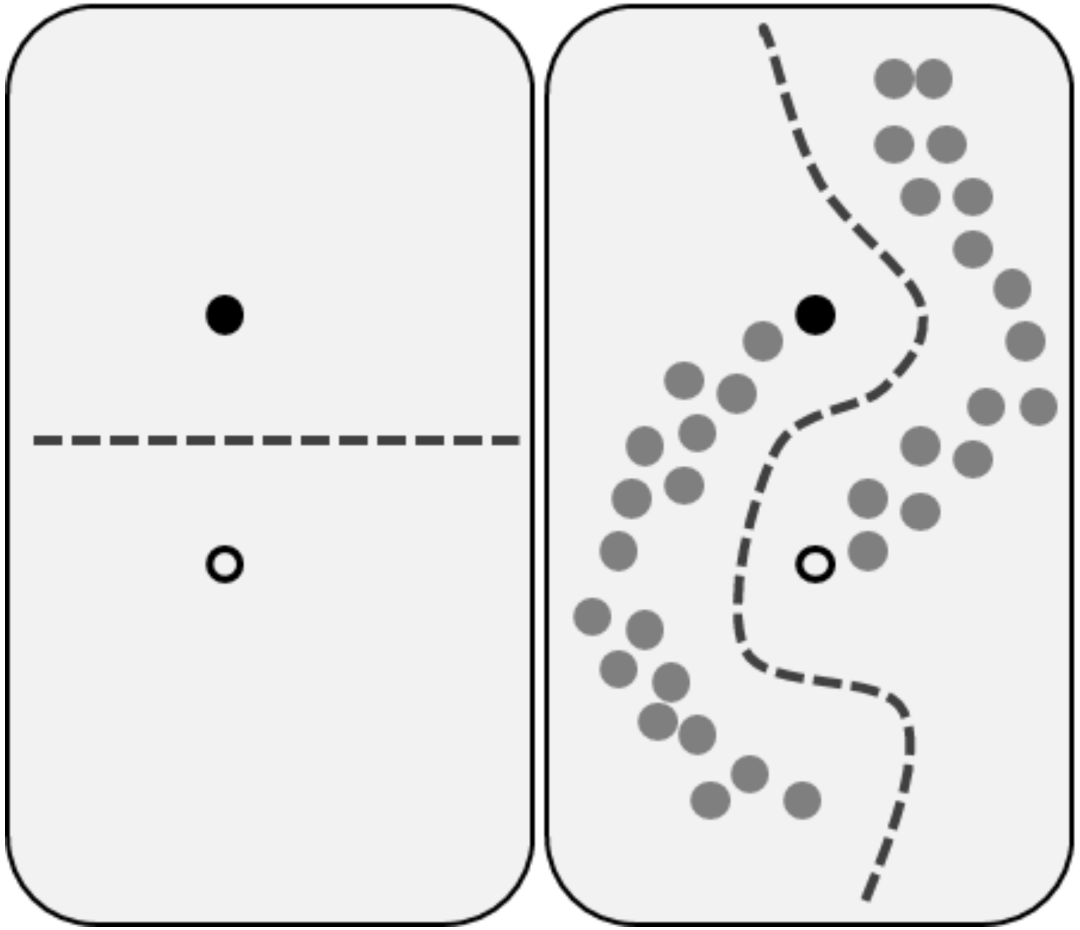
- Source set is not representative
  - Different data distributions (target vs source set)
  - Different types of TCEs in source and target sets: some sorts of bias in selecting TCEs to label first by TCERT
  - Poor features: extracted features were not good to categorize cases in the target set
- Poor Labeling
  - Target set is not labeled thoroughly and only a quick disposition is provided.

- Or a combination of both
  - We already know that labeling is a problem.
  - How to check the distribution of 216 dimensional data: studying the distance of TCEs from the decision boundary



Semi-Supervised Learning (SSL)

- Combines both labeled and unlabeled cases to obtain a better sense of the distribution of data
- Learn a decision boundary that
  - Not only separates examples in the labeled set (here source set)
  - But also passes through less dense area in the unlabeled (target) set (i.e. it considers the distribution of the data)
- Useful for our problem if differences in data distribution is a problem



Results of SSL (Q1-Q12)

- Random Forest
  - Accuracy: 0.52

Source (random forest)			
	PC	AFP	NTP
PC	2843	8	28
AFP	98	271	24
NTP	25	12	11267

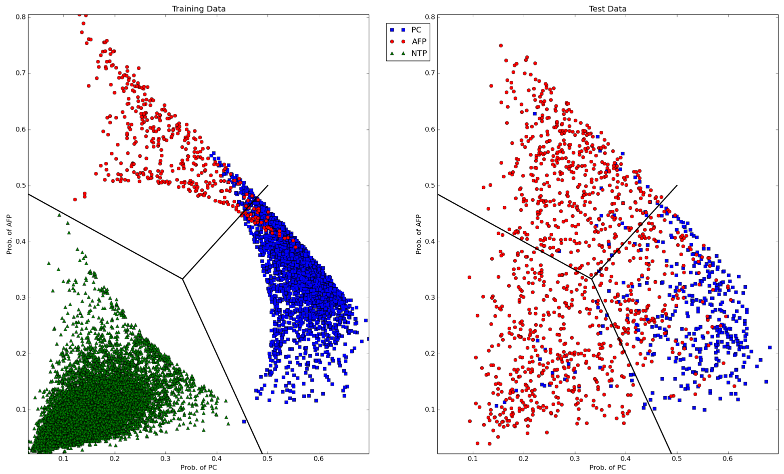
Target (random forest)			
	PC	AFP	NTP
PC	336	10	43
AFP	204	388	506
NTP			

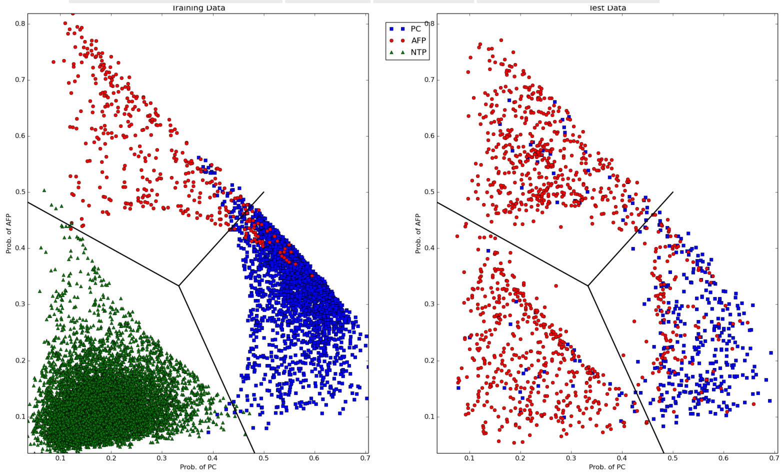
Better Models

- SSL
  - Accuracy: 0.6

Source (SSL)			
	PC	AFP	NTP
PC	2778	98	3
AFP	37	353	3
NTP	6	4	11294

Target (SSL)			
	PC	AFP	NTP
PC	312	41	36
AFP	117	529	452
NTP			



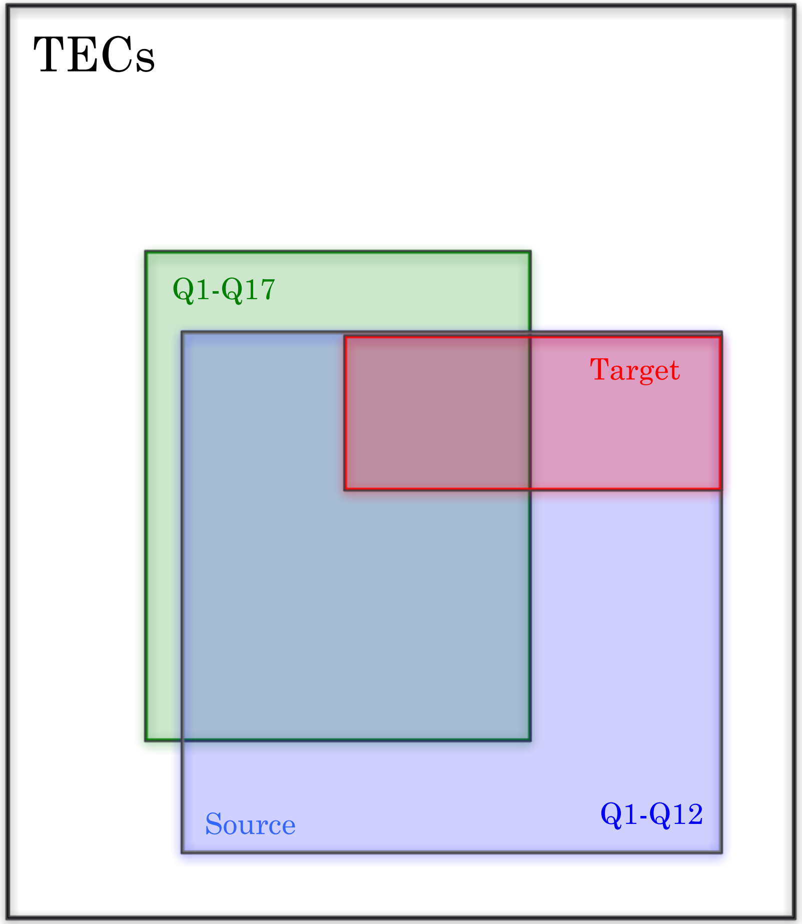


Part of the problem was differences in data distribution

With the new Q1-Q17 data set, we got better insight into what was going on

Q1-Q17 Data

- Q1-Q17 data and labels (Last data release, No.25)
- 211 features from 17 quarters of observations
- Total number of TCEs: 32k
  - Labeled TCEs: about 10k

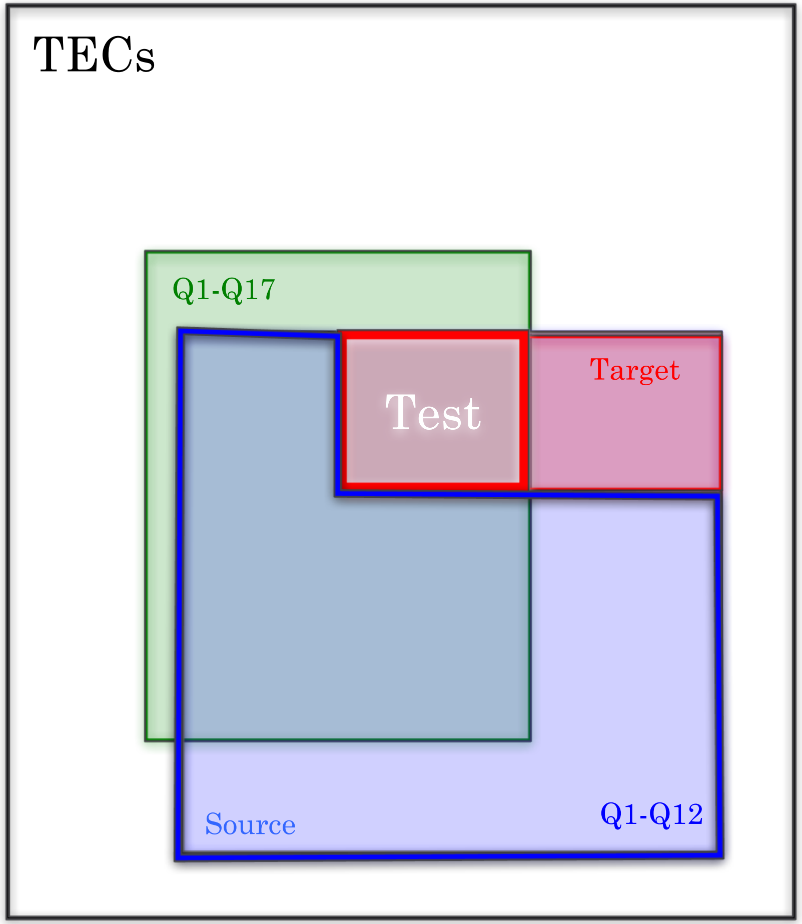


Experiment 1

- Same attributes as before
  - Training set: Similar cases and similar features as before from Q1-Q12 (Dark Blue area)
  - Test Set: A subset of target cases for which we now have the correct labels (Red Area)

- Results:
  - Accuracy: 0.60
  - Shows that labeling was a small problem

Target			
	PC	AFP	NTP
PC	301	11	16
AFP	219	381	50
NTP	1	2	15

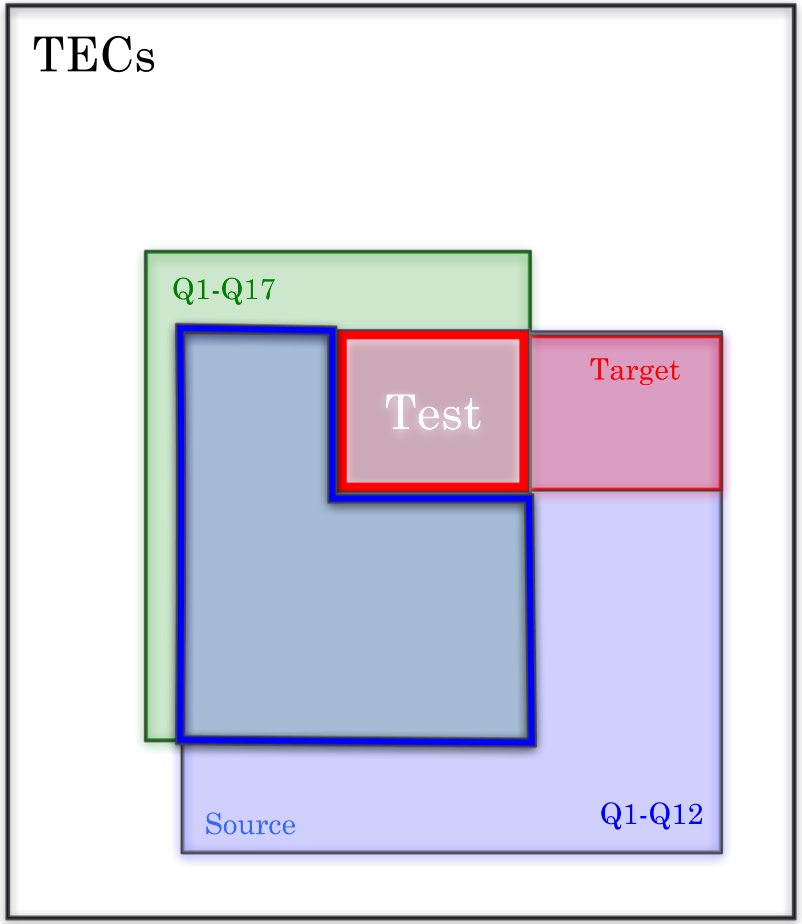


Experiment 2

- Same attributes as before
  - Training set: Q1-Q17 features and labels in DR. 25 (Dark Blue area)
  - Test Set: A subset of target cases for which we now have the correct labels (Red Area)

- Results:
  - Accuracy: 0.924

Target			
	PC	AFP	NTP
PC	300	25	3
AFP	29	761	27
NTP	1	3	14

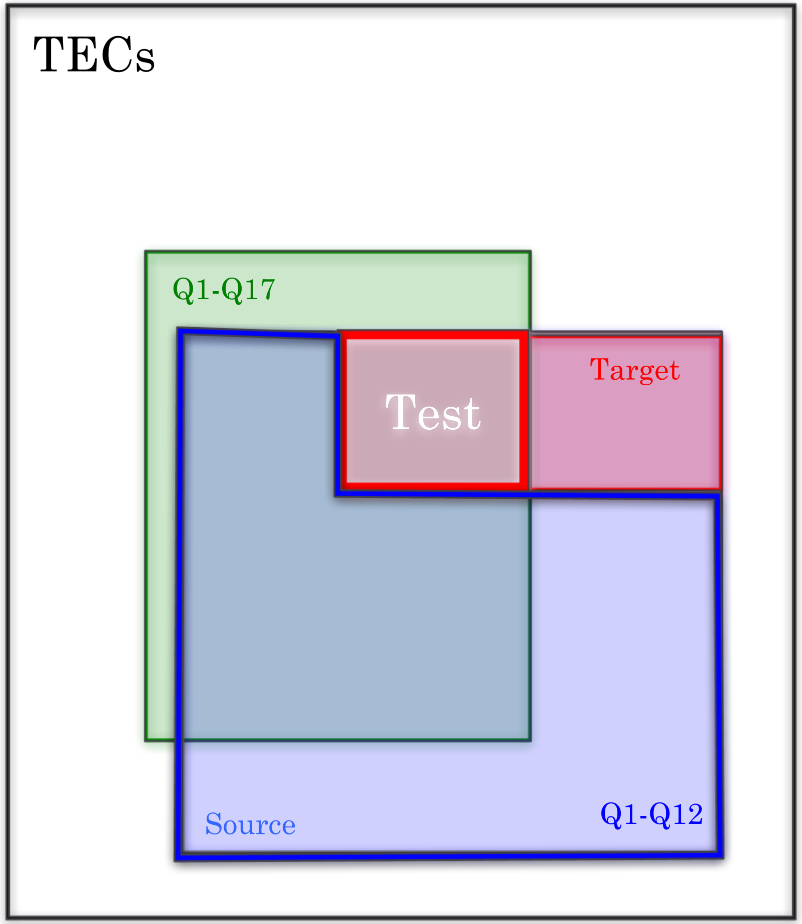


Experiment 3

- Same attributes as before
  - Training set: Q1-Q17 features and labels + unlabeled Q1-Q17 cases with labels provided from Q1-Q12 (Dark Blue area)
  - Test Set: A subset of target cases for which we now have the correct labels (Red Area)

- Results:
  - Accuracy: 0.893

Target			
	PC	AFP	NTP
PC	296	27	5
AFP	36	731	50
NTP	1	3	14



Insights from Q1-Q17

- Labeling was a problem in both target and source set in the Q1-Q12 data set

- The bigger problem was poor features
  - More reliable feature computation from 17 quarters compared to 12 quarters
  - Also, turned out that the Kepler team learned to compute/extract features over time
  - How about now: do we have good enough features? Probably not!!

- Future direction: automatic feature extraction + model construction
  - Instead of relying on the extracted features, do deep learning on the original light curve data
  - The hope is that we don't sacrifice performance by missing some important features

Summary

- Machine Learning can be used for both space engineering and science problems
  - each has different characteristics and demands different treatments
- Two case studies of machine learning in space research projects
  - Hubble Space Telescope as an extreme small sample size shows the importance of using domain knowledge
  - Using Kepler mission data, we show (1) how the lack of enough data can be compensated by utilizing the information in unlabeled data, (2) features might not be reliable. Better to use the original raw data if possible.
- Using examples, showed that machine learning process is not a pure science: needs time and resources to obtain satisfactory results.
- Visualization plays a major role in diagnostic and better understanding of the problem and how to proceed

Thank you!

Hamed Valizadegan, PhD
hamed.valizadegan@nasa.gov